Title: An Algorithm for Predicting the Arrival Time of Mass Transit Vehicles Using Automatic Vehicle Location Data

Authors: Z. Wall, D. J. Dailey
An Algorithm for Predicting the Arrival Time of Mass Transit Vehicles Using Automatic Vehicle Location Data.

Z. Wall, D.J. Dailey
University of Washington
Department of Electrical Engineering
Box 352500
Seattle, Washington  98195-2500

206-543-5371
fax 206-616-1787
zwall@u.washington.edu

206-543-2493
fax 206-616-1787
dailey@its.washington.edu

ABSTRACT

This paper presents an algorithm for predicting the arrival time of transit vehicles using a combination of both AVL and historical data. The algorithm is presented in its two components: tracking (using a Kalman filter framework) and prediction (using statistical estimation). The algorithm produces an estimate of the predicted arrival time for a given transit vehicle and provides a measure of the prediction’s accuracy.

KEYWORDS
Kalman filters, transit vehicles, arrival time prediction, APTS, automatic vehicle location system

ACKNOWLEDGMENT

This work was support in part by a series of contracts with the Federal Highway Administration and the Washington State Department of Transportation.

1. INTRODUCTION

Under the rubric of Advanced Public Transportation Systems (APTS), a number of projects have been implemented to improve distribution of pertinent information (departure time, vehicle delay, vehicle position) about a mass transit system directly to the rider. This paper presents an algorithm developed in an APTS project whose primary objective was twofold: 1) development of real-time departure information displays for transit vehicles and 2) use of such displays to provide riders waiting at transit centers with useful information. This paper discusses the development of an algorithm to accurately predict arrival times of vehicles given both real-time data on a vehicle’s position and information about its path.

2. OVERVIEW

The goal of the algorithm presented here is to accurately predict transit vehicle arrival times up to an hour in advance. Beyond the primary goals, there is an additional set of constraints on the algorithm that are imposed to facilitate implementation of the algorithm in real-world systems. These additional constraints are: 1) the uncertainty in the arrival-time must be quantified, 2) the output of the algorithm must be synchronous, and 3) lost or delayed data must be handled efficiently. Our prediction method is comprised of two sequential components, as shown in Figure 1. The first step is to estimate the current position of the bus. The second step is to use the position estimate to predict the arrival time. The position-estimation component is a tracking problem, and the travel-time prediction component is a statistical estimation problem.
3. DESCRIPTION

3.1 Tracking Using a Kalman Filter

One solution for tracking a moving object using noisy data is a Kalman filter. In this section, we present a tracker that uses the Kalman filter framework. Tracking a transit vehicle using a Kalman filter is standard practice (1); however, the Kalman filters in these systems rely on regular position and/or velocity measurements. Unfortunately, the asynchronous (and often sporadic) data feed from our target AVL system does not meet the specifications used in the earlier models. In this work, we generate a new model that takes into account the nature of an AVL system that samples position irregularly.

Our model for tracking the vehicle assumes that all of the secondary effects (weather, ridership, congestion, etc.) show up as noise in the trajectory. Unlike the reference system, our model does not assume rapid (e.g. one sample a second) synchronous sampling; rather, it allows for the frequency of the measurements to be much lower (e.g. one sample a minute). The error caused by ignoring the secondary effects is much smaller in proportion to the error from the undersampling. Further, since we are only tracking the position of the bus in one-dimension (the distance-until-destination axis), the error does not cause the bus to go off the route. By ignoring the intricate details of the physical motion of the vehicle, the description of the model of the bus becomes more a functional model than a physical model.

The key to this functional model is the relationship between the transit vehicle and driver. The vehicle does not move without the driver controlling it. Therefore, we model the physical behavior of the bus as a deterministic system with a control input and stochastic error. We do not have a direct sensor for the control input of the bus, but we are able to approximate the control input because the driver does not drive arbitrarily. Instead, the driver’s control input conforms to a set schedule. The schedule (as delivered to the driver) consists of link descriptions and corresponding travel-times for those links. From this, the approximate control input is the particular velocity over different regions of the route. Accordingly, if we generate a profile of the velocity using data recorded from previous trips, we can use this profile as the control input.

This is the process model for the vehicle. Our measurement model relates the distance-until-destination measurement, the only sensor available, and the distance-until-destination state variable. Having defined the process and the measurement model, we pose our problem in the Kalman filter framework. The general forms of the process [1] and measurement [2] equations for a scalar Kalman filter are shown below. (2)

\[
x(k) = ax(k - 1) + qu(k - 1) + w(k - 1) \\
y(k) = cx(k) + v(k)
\]
Where:

1. The variable $x$ tracks the distance-until-destination.
2. The process coefficient $a$ is the identity matrix.
3. The measurement coefficient $c$ is also the identity matrix.
4. The control input $u$ is a velocity.
5. We set the control parameter $\varphi$ to the time step.
6. $w(k)$ is the error in the process model.
7. $v(k)$ is the error in the measurement model.

This results in the following definitions:

\[ a = 1 \quad c = 1 \quad \varphi = \tau \quad (3) \]

Now that we have defined the process and measurement model, the next step is to describe the properties of the noise sequences $v(k)$ and $w(k)$. Since the process model is deterministic, it does not contribute to the errors in the process equation. However, since we estimate the velocity profile from sampled data, it has errors associated with it. In fact, since the contribution from the deterministic portion is zero, we take $w(k)$ to be the error in our estimate of the velocity. We model this error as a normal random variable with zero mean and a known variance. The variance value used with the filter is the result of the linear regression calculation used to generate the velocity from the previous trip data.

In the same way, we define the error sequence $v(k)$ to be the error in the AVL positioning system. This error sequence is also a normal random variable with zero mean and a known variance. The information on the variance and the error process is provided by the developer of the AVL system and represents the performance of the system when it is working correctly. Additional steps can be taken to ensure that this is the case. For this paper we assume that the system is working correctly.

In order to finalize this method, we look at the filter solution for two different cases: 1) no new data is received or 2) new data is received. In the absence of new data, we make an estimation of the current state using only the process model. We know that the uncertainty in our estimation is growing, and we can account for this by increasing the error by the variance of the noise. This results in the following set of equations (3):

\[
\begin{align*}
x(k) &= ax(k-1) + \varphi u(k-1) \\
p_i(k) &= a^2 p(k-1) + \sigma_w^2 \\
p(k) &= p_i(k)
\end{align*}
\]

where $p(k)$ is the error sequence.

In the other scenario, where new data is received, we use the scalar filter solution, from (3) to update the position sequence using both the last estimated position and a current position measurement. This results in,

\[
\begin{align*}
b(k) &= cp_i(k) [c^2 p_i(k) + \sigma_v^2]^{-1} \\
x(k) &= ax(k-1) + b(k)[y(k) - axc(k-1)] \\
p_i(k) &= a^2 p(k-1) + \sigma_w^2 \\
p(k) &= p_i(k) - cb(k)p_i(k)
\end{align*}
\]

where $b(k)$ is the gain and $p(k)$ is the error sequence.
The result of these equations is the optimum blend (in the least squares sense) of the prediction (from the process model) and the current measurement of the distance-until-destination (from the AVL data). This results in the minimum variance estimate of the distance-until-destination state. Besides the minimum variance guarantee, the other key result is that the tracker still functions without a steady stream of measurements. In this situation, the errors that result from the lack of current data are reflected by a corresponding increase in the error sequence generated by the filter.

3.2 Prediction Component

The second stage of the algorithm is the prediction component. The prediction component uses three pieces of information to calculate the arrival time: the current time, the current estimate of the distance-until-destination, and a travel-time function for the route that the transit vehicle is on. After one step of the Kalman filter, we have a measure of the current time and the distance-until-destination estimate. Accordingly, the next step is to use the travel-time function to calculate the time required to reach the destination from the current position. To use this methodology, a distribution of travel times is needed.

For this work, we define a travel-time-function to one that relates the distance a transit vehicle has to travel with the time it takes to travel that distance. The travel-time function provides a way to ‘convert’ the distance-until-destination estimate from the Kalman filter into an associated time-until-arrival estimate. We define the ‘time-until-arrival’ as being the time in minutes (or hours) that it takes the bus to travel from its current position to the goal. Defining the function this way allows the data to be manipulated in a relative manner. The function is not tied to absolute times or distances; instead, it relates specific distance/travel-time pairing.

In addition, the data contains noise associated with perturbations resulting from physical effects: the bus missed a stoplight, heavy congestion in a particular area, bad weather, etc. These physical effects contribute to the overall delay, and this delay is present in the measurement. Moreover, the further the bus has traveled, the more likely it is to have experienced some delays. While the absolute measurement contains all of the delays over the entire trip, the relative measurement removes the effects of all the delays prior to that point, leaving only the future delays. In this way, the relative measurement encapsulates the delays yet come.

This is important because at the time of prediction we already have a measurement of current time and the current distance-until-destination. Combining these together, we know the previous delays. The absolute measurement gives us a combination of past and future delays, and the relative measurement isolates the future delays.

We develop the travel-time function using statistical methods. Using historical data comprised of individual \{distance-until-destination, travel time\} pairs collected from previous runs of the same trip, we portray the relationship between the distance-from-destination and the travel-time as the joint density function.

An example of the historical data collected from a single bus trip is shown in Figure 2. In this figure, the x axis is the distance-from-destination, while the y axis is the travel-time. The data (as sampled) suggests that they are not in the correct form for use in the prediction process as the data represents a-priori information about a trip.

We cannot use the historical data alone to drive the prediction process. We can, however, combine our current distance-until-destination estimate with the data to isolate a specific subsection. We accomplish this by approximating the conditional density \(f(y | x)\). This conditional density represents the probability of a particular travel-time \(y\) given a particular distance-until-destination estimate \(x\). Moreover, at any distance along the route, there is a corresponding distribution of potential travel-times. This distribution is continuous and represents the probability of particular travel-times occurring with perturbations caused by unobservable noise processes. Also, we assume that this distribution is normal.

At this point, we have the basis of a functional relationship between distance-until-destination and travel-time in that, given a particular distance, we can find an associated travel-time distribution from the historical data and calculate statistics from it (such as the mean travel-time). The estimate from the Kalman filter has an associated variance. We therefore expand our definition of conditional density to include a range of distances.
Using the general definition for a conditional density (see 7-40 in (4)), we see that the density over a range is composed of the marginal density in the range (see 6-6 in (4)) scaled by the probability of the range. From our initial observation that each distance has a travel-time distribution associated with it, we conclude that the conditional density for a range of positions is the convolution of the individual densities for each position for the range. The approximate conditional density extracted from the historical data serves as the basis for our functional relationship between distance-until-destination and travel-time. The first step in the approximation process is to choose the appropriate standard density function to approximate the conditional density. We assume that the density is normal. This is a reasonable assumption based on the central limit theorem.

To estimate the mean and variance of the density, we first assume that samples of runs of the same trip taken on different days represent realizations of the same trip. The schedule dictates a particular trajectory for a particular trip. Samples of this trip (even if they are from different runs) represent a realization of the trajectory defined by the schedule. We assert that these travel-time measurements also represent realizations of the travel-time density conditioned upon the distance associated with them. This assertion results from the earlier concept of every distance along the trip having an associated travel-time distribution.

From realizations contained in \{distance-until-destination, travel-time\} pairs, we calculate the sample mean and variance using the following formulae,

\[ \eta = \frac{1}{N} \sum_{i=1}^{N} \text{traveltime}_i \]  \hspace{1cm} (11)

\[ \sigma^2 = \frac{1}{N-1} \sum_{i=1}^{N} (\text{traveltime}_i - \eta)^2 \]  \hspace{1cm} (12)

where the range determines the index \( i \).

To incorporate the uncertainty in the distance measurement, we generate a 90% confidence interval. Using the bounds of the 90% confidence interval as the range, we estimate the conditional probability density. This new probability density describes the probable travel-times for all distances in the range. We can predict the most likely
travel-time of the bus by calculating the expected value of the probability density. Using the variance of the density, we can determine the confidence interval for travel-time. This gives us an insight into the quality of the prediction.

To test the validity of the distribution membership of the data, we use a standard distribution membership test, the Kolmogorov-Smirnov (K-S) test (5). The K-S test computes the probability that two distributions are the same. The K-S test uses a metric of maximum absolute difference between two cumulative distribution functions. The metric is used in the computation of the following sum (5),

\[
Q_{KS}(\lambda) = 2 \sum_{j=1}^{\infty} (-1)^{j-1} e^{-2j^2 \lambda^2}
\]

\[
\lambda = \left[ \sqrt{N_e} + 0.12 + 1.11 / \sqrt{N_e} \right] D
\]

Larger K-S test values show that the two cumulative distributions are similar. To apply the K-S test to our data, we use the following steps:

1. Select an appropriate range of samples from the joint density of a single trip.
2. Find the sample mean and variance of these points.
3. Generate a normal distribution using the calculated mean and variance.
4. Use the selected points to create an unbiased estimator of actual distribution.
5. Compare the two distributions to find the maximum absolute distance (the K-S Statistic).
6. Use the K-S statistic to compute the probability that the selected points came from the approximate distribution.

Figure 3 shows the results of this test. It shows the K-S probability for a variety of distances on a single trip. For this test, the range was fixed at a size corresponding to the 90% confidence interval for a position estimate with a standard deviation of 75 meters. Although at certain times the probability drops below 80%, the probability ranges between 80% - 99%. From these tests, we conclude that this method of approximating the conditional density has a high probability of being accurate.

Figure 3. Result of the Kolmogorov-Smirnov Test
From these experiments, we can conclude that by approximating the conditional distribution over a range as normal and deriving the mean and variance from the samples, we can get an accurate picture of the travel-time for a particular distance-until-destination estimate. Once we have this travel-time, we use it to predict the arrival-time using the following formula,

\[ \text{arrival-time} = \text{current-time} + \text{travel-time} \tag{15} \]

4. EMPIRICAL RESULTS

In this section, we present the empirical results for the prediction algorithm. We evaluate the individual performance of the tracking component and the overall performance of the algorithm. In order to demonstrate the flexibility of the system, we choose an arbitrary bus run and evaluate the performance for both normal and sparse data streams.

We use two different datasets taken from the same bus. The datasets were collected on different days but at the same time of day. Dataset-1 is the result of a regularly sampled data collection. Dataset-2 is the result of a sparsely sampled data collection session. These two different modes of operation result from limitations in the sampling method used by the AVL system. The AVL system can only poll twelve buses every second. As a result, in situations where a large number of buses are being tracked simultaneously or a select few buses are being tracked very closely by human operators, some buses are not polled at a regular interval (their polling position is used to poll a different bus). This results in the sparse and normal data cases which are representative of the data generated by the AVL system. Using these datasets, we reach several significant conclusions.

The first set of empirical results presented here is the performance of the tracking component. Figure 4 shows the performance of the tracking algorithm when using dataset-1 (normal), and Figure 5 shows the performance of the tracking algorithm when using dataset-2 (sparse). In each figure, the solid line represents the trajectory as estimated by the tracking component (Kalman filter), and the points marked with an ‘x’ are the actual data samples.

![Figure 4. Performance of the tracking component using dataset-1 (normal)](image-url)
We find that the tracking component continues to function well even in situations where data is sparse. For example, in Figure 5, between times 21.3 and 21.6 on the y-axis (18 minutes), only two samples are received. Despite this lack of data, the process model in the filter is still able to generate acceptable estimates for the position. We validate this assertion by performing a sanity check on the magnitude of the average velocity over each segment of the estimated trajectory. In general, the magnitude of the velocity remains near the speeds the bus actually travels. When the process model is functioning correctly, there are no dramatic corrections in the when a new measurement is received. We define a dramatic correction to be a large distance change (greater than 1500 meters) during a single timestep (30 seconds). In physical terms, travelling 1500 meters within a thirty-second interval requires a velocity that a bus is not capable of travelling at.

![Figure 5. Performance of the tracking component using dataset-2 (sparse)](image)

To examine the prediction component, we compare the predictor with the observed arrival time. Figure 6 shows the prediction results (in decimal hours) versus the clock time when the prediction was made for both datasets. In this figure, the solid horizontal line is the observed arrival time, the predicted arrival times are shown as ‘x’s’, and the 90% confidence intervals are indicated by dotted lines.

![Figure 6. Comparison between the predicted arrival time and the actual arrival time](image)
From this, we find that the algorithm is able to produce accurate (within 5 minutes) predictions when either normal or sparse data is present. As shown by Figure 7, the error in the prediction is less than five minutes in all cases with a majority of the predictions being within two minutes of the actual arrival time. While this result is only for two trips, later results show that this behavior carries on across a large number of tests.

![Normal Data Rate](image1)

![Sparse Data Rate](image2)

Figure 7. Error in the predicted arrival time

Having shown the empirical results for a single trip, we now expand the analysis to cover a larger number of trips. In order to do this, we present results based upon specific statistics taken from a number of bus routes at different times. These results represent the behavior of over 100 different datasets drawn from a large pool of different runs.

From this analysis, we can assess quality and usefulness of the algorithm. From rider information surveys (6), the average wait time for transfer riders is 15.9 minutes. Assuming that non-transfer riders use a printed schedule to minimize their wait times, we assert that more than a majority of riders will spend less than 16 minutes at the bus stop. As such, we test the accuracy of the algorithm against the earliest time a typical rider at a bus stop would want to know arrival-time information - 15 minutes before the bus arrives. Now, we create an unbiased estimator of the cumulative distribution (shown in Figure 8) of the magnitude of the error in the predicted arrival time. The estimator represents the probability distribution that predictions made 15 minutes before the actual arrival time of the bus will have an associated error of less than a given amount, as discussed below.

As shown in Figure 8, the magnitude of the error is less than 2 minutes with probability .7. If even more certainty is required, the magnitude of the error is less than 4 minutes with probability .9. The rider information survey states that ‘Notably, regular and infrequent riders are less satisfied with their ability to get information about Metro by telephone, time between buses, and safety while riding the bus after dark’(6). The algorithm improves two of these three areas: 1) by providing up-to-date information directly to the rider in a timely manner and 2) clearly defining the wait time between buses to reduce the associated dissatisfaction. Given that the percent error in the travel-time estimate is less than 12% (with P = .7) 15 minutes before the bus arrives and that the accuracy of the estimates improve as the bus gets closer, the algorithm should improve the rider’s experience in the aforementioned areas.
5. CONCLUSION

This paper presents an algorithm to predict the arrival time of transit vehicles. The algorithm combines real-time AVL data with an historical data source to produce a distribution of travel times. Using this distribution, the algorithm calculates the expected travel-time. The predicted arrival time is the sum of the expected travel-time and the current time. The distribution is then used for evaluating the reliability of the estimate. Empirical results have shown that the proposed algorithm is flexible enough to function in adverse conditions and is able to produce predictions that are useful to the rider.

REFERENCES